

On page 43, line 17, delete "snapshot 2 and for blkmap file 2344." and insert --blkmap file 2344 and snapshot 2. Buffer 2308 is already marked with an asterisk because inodes 2308A and 2308B are dirty in Figure 21A-- in place thereof.

On page 43, line 19, insert --as illustrated in Figure 21B, where block 2326 is marked with an asterisk-- after "dirtied".

On page 44, line 9, insert --, as indicated in Figure 21F-- after "disk".

On page 44, line 31, insert --be-- after "to".

On page 45, lines 21-22, delete "2433 of snapshot 2432 that references it" and insert --at which direct block 2410 was accessed-- in place thereof.

#### IN THE DRAWINGS

Please amend Figures 17G, 21B, and 23B as shown in the accompanying drawing sheets, in which the proposed amendments are indicated in red ink.

#### IN THE CLAIMS

Please amend pending claims 1 and 2 as follows.

1. (Once amended) A method for [generating a] maintaining a file system stored in non-volatile storage means at successive consistency [point] points, said file system comprising blocks of data, said blocks of data comprising blocks of regular file data and blocks of meta-data file data referencing said blocks of data of said file system, said meta file data comprising a file system information structure comprising data describing said file system at a first consistency point, said computer system further comprising memory means, said method comprising the steps of:

maintaining a plurality of modified blocks of regular file data and meta-data file data in said memory means, said modified blocks of data comprising blocks of data modified from said first consistency point;

[marking a plurality of inodes pointing to a plurality of] designating as dirty blocks of meta-data file data referencing said modified blocks of regular file data and meta-data file data [in a file system as being in a consistency point], said dirty blocks of meta-data file data comprising blocks of meta-data file data to be included in a second consistency point:

[flushing] copying said modified blocks of regular [files] file data referenced by said dirty blocks of meta-data file data to free blocks of said non-volatile storage means;

[flushing special files] copying blocks comprising said modified blocks of meta-data file data referenced by said dirty blocks of meta-data file data to free blocks of said non-volatile storage means;

modifying a copy of said file system information structure maintained in said memory means to reference said dirty blocks of meta-data file data;

[flushing at least one block of] copying said modified file system information structure to said non-volatile storage means; and,

requeueing any dirty inodes that were not part of said consistency point].

2. (Once amended) The method of claim 1 wherein said blocks of meta-file data comprise one or more blocks of inode file data and one or more blocks of blockmap file data and wherein said step of [flushing] copying said [special files] modified blocks of meta-data file data to free blocks of said non-volatile storage means further comprises the steps of:

[pre-flushing] copying an inode [for a] referencing one or more blocks of blockmap file data to a block of [an] inode file data maintained in said memory means;

allocating [space on] free blocks of said non-volatile storage means for [all dirty blocks in] said block of inode file data and one or more modified blocks of blockmap [files] file data;

updating said inode referencing said one or more blocks of blockmap file data to reference said one or more free blocks of said non-volatile storage means allocated to said one or more modified blocks of blockmap file data;

[flushing] copying said updated inode [for said blockmap file again] to said block of inode file data;

updating [a plurality of entries in] said one or more blocks of blockmap file data [wherein each entry of said plurality of entries represents a block on said storage means; and,];

writing [all dirty blocks in said] said updated one or more blocks of blockmap file data and said block of inode file data to said allocated free blocks of said non-volatile storage means.

Please add the following new claims 3-21:

--3. A method for maintaining a file system comprising blocks of data stored in blocks of a non-volatile storage means at successive consistency points comprising the steps of:

storing a first file system information structure for a first consistency point in said non-volatile storage means, said first file system information structure comprising data describing a layout of said file system at said first consistency point of said file system;

writing blocks of data of said file system that have been modified from said first consistency point as of the commencement of a second consistency point to free blocks of said non-volatile storage means;

storing in said non-volatile storage means a second file system information structure for said second consistency point, said second file system information structure comprising data describing a layout said file system at said second consistency point of said file system.--

--4. The method of claim 3 wherein said step of storing said first file system information structure in said non-volatile storage means comprises the step of:

storing first and second copies of said first file system information structure at first and second locations respectively of said non-volatile storage means;

and wherein said step of storing said second file system information structure in said non-volatile storage means comprises the steps of:

overwriting said first copy of said first file system information structure with a first copy of said second file system information structure; and

overwriting said second copy of said first file system information structure with a second copy of said second file system information structure.--

--5. The method of claim 4 wherein said first and second locations of said non-volatile storage means comprise fixed predetermined locations of said non-volatile storage means.--

--6. The method of claim 4 wherein each copy of said file system information structure comprises means for determining a most recent version of said file system information structure and means for determining validity of said file system information structure, further comprising the steps of:

after a system failure, reading said first and second copies of said file system information structure from said first and second locations of said non-volatile storage means;

determining a most recent valid file system information structure from said first and second copies of said file system information structure.--

--7. A method for creating a plurality of read-only copies of a file system stored in blocks of a non-volatile storage means, said file system comprising meta-data identifying blocks of said non-volatile storage means used by said file system, comprising the steps of:

storing meta-data for successive states of said file system in said non-volatile storage means;

making a copy of said meta-data at each of a plurality of said states of said file system;

for each of said copies of said meta-data at a respective state of said file system, marking said blocks of said non-volatile storage means identified in said meta-data as comprising a respective read-only copy of said file system.--

--8. The method of claim 7 wherein said step of marking said blocks comprising a respective read-only copy of said file system comprises placing an appropriate entry in a means for recording multiple usage bits per block of said non-volatile storage means.--

--9. The method of claim 8 wherein said means for recording multiple usage bits per block of said non-volatile storage means comprises a blockmap comprising multiple bit entries for each block.--

--10. The method of claim 7 wherein said meta-data comprises pointers to a hierarchical tree of blocks comprising said file system.--

--11. The method of claim 7 wherein said meta-data comprises structures representing files of said file system.--

--12. The method of claim 11 wherein said structures representing files of said file system comprise inodes.--

--13. The method of claim 7 further comprising the step of:

preventing overwriting of said blocks marked as belonging to a read-only copy of said file system.--

--14. The method of claim 7 comprising the step of unmarking said blocks marked as belonging to a read only copy of said file system when said read only copy of said file system is no longer needed.--

--15. The method of claim 7 wherein a plurality of said blocks marked as belonging to a read-only copy of said file system comprise data ancillary to said file system, said method further including the steps of:

allowing said ancillary data to be overwritten; and  
otherwise preventing overwriting of said blocks marked as comprising a read only copy of said file system.--

--16. The method of claim 15 wherein said ancillary data comprises access time data.--

--17. The method of claim 7 wherein said meta-data comprises a root structure referencing structures representing files of said file system, and wherein said copies of said meta-data comprise copies of said root structure.

--18. The method of claim 17 wherein said root structure comprises a root inode.--

--19. The method of claim 7 further comprising the step of using one or more of said read-only copies of said file system to back-up said blocks comprising one or more consistency points of said file system.--

--20. A method for recording a plurality of data about a plurality of blocks of data stored in storage means comprising the steps of: